AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A method comprising:

generating a first plurality of message digests that correspond to a first plurality of file contents on a client connected with a network, wherein the first plurality of message digests uniquely identify the first plurality of file contents ~~contents of files stored on the client~~;

generating a second plurality of message digests that correspond to a second plurality of file contents on a repository connected with the network, wherein the second plurality of message digests uniquely identify the second plurality of file contents;

combining the first plurality of message digests into a single client message digest;

combining the second plurality of message digests into a single repository message digest;

comparing the single client message digest with the single repository message digest to determine file contents that do not match; and

synchronizing the file contents that do not match with the client and the repository ~~contents of the client with a repository connected with the network based on contents of the message digests on the client and corresponding entries in a database of message digests stored on the repository; verifying that the contents of the repository match the contents of the client; and marking those contents of the client that did not match the contents of the repository for later copying to the repository~~.

2. (Currently Amended) The method of claim 1, further comprising storing the first plurality of message digests on the client.

3. (Currently Amended) The method of claim 2, further comprising generating a new plurality of message digests for ~~the~~ files on the client to be cached on the repository prior to synchronizing ~~data synchronization~~.

4. (Currently Amended) The method of claim 1, wherein the first plurality of file contents comprises ~~files stored on the client comprise~~ a subset of ~~the~~ files stored on the client.

5. (Canceled)

6. (Currently Amended) The method of claim 1, wherein the generating of the first and second plurality of message digests comprises generating a cryptographic hash for each file content to be synchronized.

7. (Previously Presented) The method of claim 6, wherein the cryptographic hash comprises 128 to 160 bits.

8-9. (Cancelled)

10. (Currently Amended) A system comprising:

a repository server connected with a network, the repository server to function as a data repository on behalf of a client,

generate a first plurality of message digests that correspond to a first plurality of file contents on the repository, wherein the first plurality of message digests uniquely identify the first plurality of file contents, and

combine the first plurality of message digests into a single repository message digest; and

the client connected with the repository server via the network, wherein the client is to

generate a second plurality of message digests that correspond to a second plurality of file contents, wherein the second plurality of message digests ~~each~~ uniquely identify the second plurality of file contents~~content of a corresponding file stored on the client~~,

combine the second plurality of message digests into a single client message digest,

compare the single client message digest with the single repository message digest to determine file contents that do not match, and

synchronize the file contents that do not match with the client and the repository~~contents of the client with files stored in the repository server based on contents of the message digests on the client and a database of message digests stored on the repository, verify whether the contents of the repository match the contents of the client, and mark those contents of the client that did not match the contents of the repository for later copying to the repository~~.

11. (Currently Amended) The system of claim 10, wherein the generating of the first and second of plurality of message digests comprises performing a cryptographic hash for each file content to be synchronized.

12. (Previously Presented) The system of claim 11, wherein the cryptographic hash comprises 128 to 160 bits.

13-19. (Cancelled)

20. (Currently Amended) A machine-readable medium having stored thereon data representing sets of instructions which, when executed by a machine, cause the machine to:

generate a first plurality of message digests that correspond to a first plurality of file contents on a client connected with a network, wherein the first plurality of message digests uniquely identify the first plurality of file contents~~contents of files stored on the client~~;

generate a second plurality of message digests that correspond to a second plurality of file contents on a repository connected with the network, wherein the second plurality of message digests uniquely identify the second plurality of file contents;

combine the first plurality of message digests into a single client message digest;

combine the second plurality of message digests into a single repository message digest;

compare the single client message digest with the single repository message digest to determine file contents that do not match; and

synchronize the file contents that do not match with the client and the repository~~contents of the client with a repository connected with the network based on contents of the message digests on the client and corresponding entries in a database of message digests stored on the repository; verify that the contents of the repository match the contents of the client; and mark those contents of the client that did not match the contents of the repository for later copying to the repository~~.

21. (Currently Amended) The machine-readable medium of claim 20, wherein the client stores the first plurality of message digests.

22. (Currently Amended) The machine-readable medium of claim 21, wherein the sets of instructions, when executed by the machine, further cause the client to generate ~~generates~~ a new plurality of message digests for ~~all~~ files on the client to be cached on the repository prior to synchronizing ~~data synchronization~~.

23. (Currently Amended) The machine-readable medium of claim 20, wherein the first plurality of file contents comprises ~~files stored on the client comprise~~ a subset of ~~all~~ files stored on the client.

24. (Cancelled)

25. (Currently Amended) The machine-readable medium of claim 20, wherein the client generates a cryptographic hash for each file content to be synchronized.

26. (Previously Presented) The machine-readable medium of claim 25, wherein the cryptographic hash comprises 128 to 160 bits.

27-28. (Cancelled)